Technical Approach for the eAuthentication Interim Capability

Draft version 0.0.1
11/03/03

Executive Summary

This document provides a description of the interim technical approach for the eAuthentication initiative. The approach is based on an architectural framework that allows multiple protocols and federation schemes to be supported over time. The approach is presented in terms of use cases. This is a pre-release draft, subject to major revision and update. Do not distribute without permission from the authors.

Table of Contents

Document History

Release	Date	Comment
0.0.1	11/3/03	Initial Draft, limited release, do not distribute

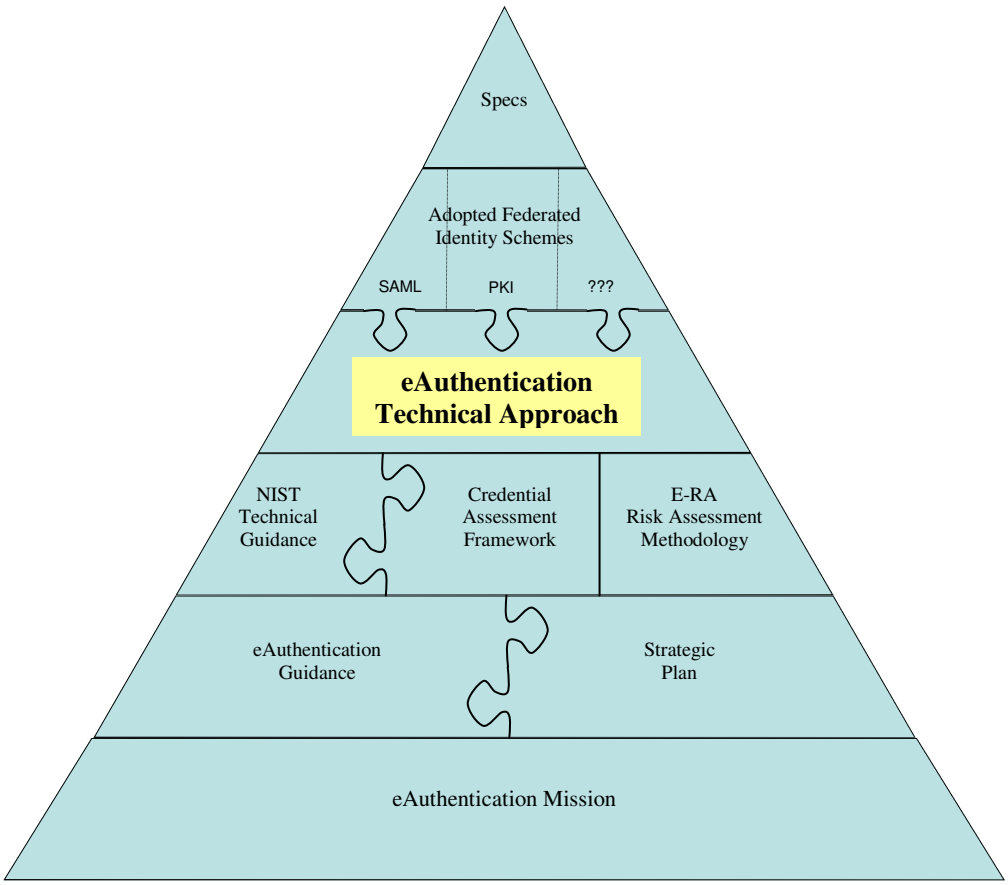
1 Introduction

1.1 Purpose of this Document

The purpose of this document is to set the technical direction and approach for pursuing a federated identity architecture under the eAuthentication initiative. It is intended to generally describe the structure under which the eAuthentication Program Management Office (PMO) will implement technologies, products and technical standards to the meet its near-term program objectives. This document further provides a methodology for the graceful adoption of new schemes as they emerge.

This document is not autonomous, but builds upon the concepts and precepts of the draft E-Authentication Guidance for Federal Agencies, NIST E-Authentication Technical Guidance, Credential Assessment Framework (CAF) and Electronic Risk Assessment Methodology (eRA). Additional technical specifications in turn build on this document. Figure 1 shows the documentation relationships for eAuthentication. The most recent version of these documents is available at the eAuthentication website, http://www.cio.gov/eAuthentication/.

Figure 1: eAuthentication Documents



1.2 The eAuthentication Concept

As part of the President's Management Agenda, the eAuthentication initiative will ultimately enable trust and confidence in e-Government transactions through the establishment of an integrated policy and technical infrastructure for identity management. Through the initiative, citizens and businesses will have simpler access to multiple agency applications through the re-use of credentials and established identities.

The eAuthentication concept is best described through the trust relationships between Agency Applications (AA), Credential Service Providers (CSP) and End-Users. CSPs are commercial or government entities authorized by the PMO to provide credentials (PINs, Passwords, Digital Certificates, etc) to potential end-users for access to government systems. AAs are government applications, systems or services that rely on (or trust) the authentication/credential services of CSPs. End-Users are people or organizations that have credentials issued by a CSP and desires to use that credential to conduct business with an AA. It is the management of transitive trust between these entities (AA, CSPs and End-users) that is the essence of the eAuthentication initiative. eAuthentication provides:

- Policies and Guidelines for federal authentication;
- Credential Assessments and Authorizations;
- Interoperability Testing of candidate products, schemes or protocols and
- Management and Control of accepted federation schemes operating within the environment.

To manage the trust relationships, the PMO does not envision building an authentication infrastructure as a central broker for these entities. Instead, eAuthentication will be a federated architecture that leverages credentials from multiple domains through certifications, guidelines, standard adoption and policies. The architecture would accommodate the use of Low-Level credentials (PIN and Passwords) as well as High-Level credentials (digital certificate and PKI) within the same environment. It is further envisioned that the architecture would accommodate emerging federation schemes such as SAML and Liberty Alliance and not simply be built around a single scheme or commercial product.

1.3 Requirements

The vision and direction for the eAuthentication initiative is contained in the eAuthentication Strategic plan. The strategic plan provides specific actionable tasks to achieve the eAuthentication Mission. The following architectural requirements are derived from the strategic plan:

High Level Requirements:

1. **Leverage**: A credential from any approved credential service should be usable at any application of equal or lower assurance level. Agency applications must be able to leverage existing credentials rather than establish new identity management systems.
2. **Single Sign-on**: Once a user has authenticated they must be able to move among applications with equivalent assurance levels without re-authenticating.
3. **Privacy**: There must be no central audit log of which users accessed which applications and no centralized identity management system. Credentialing must be federated among multiple providers.
4. **Governance**: The architecture must provide for explicit control over which applications and credential services can join the eAuthentication community.

Design Goals:

1. **Standards**: The architecture should rely on existing industry standards.
2. **COTS**: The architecture should employ COTS products.
3. **Federation**: Authentication should be federated among multiple credential providers
4. **Durability**: The architecture should be designed to allow for the evolution of technology, providing for easy migration as the industry evolves.
5. **Flexibility**: The architecture should not create undue reliance on any single standard, vendor, product, or integrator.

2 Approach

The technical approach for eAuthentication is based on an architectural framework that allows for the co-existence of multiple federated identity schemes within a single architecture. The framework includes a methodology and process for the evaluation and adoption of these schemes over time. The goal of the framework is to provide a lasting architectural model for eAuthentication that is not irrevocably bound to a single industry standard, vendor, or product.

The approach is presented though use cases depicting the high level interaction of eAuthentication components in various scenarios. The major sub-components of eAuthentication are:

1. **Agency Applications (AAs):** eGovernment applications that perform some business function online. AAs manage all business transactions and end user authorization decisions. One of the principle goals of the eAuthentication initiative is to provide broad authentication services to AAs, allowing the complete deferral of identity management.
2. **Credential Services (CSs):** Services which provide end-users with credentials that can be used at eAuthentication-enabled AAs. CSs are provided by Credential Service Providers (CSPs)[1], which are companies that operate one or more CSs.
3. **eAuthentication Portal**: A website that helps users locate the CSs and AAs they need to complete their transactions. The Portal also maintains information about CSs and AAs referred to as metadata, which includes technical interface data as well as descriptive information.
4. **End Users**: Any citizen, government employee, contractor, or business who uses an AA. One of the principle goals of eAuthentication is to make the End User experience as simple as possible by improving the availability and ease of use of credentials.

Within the framework the End User interacts directly with AAs, CSs, and the Portal. Typically the user starts at the portal in order to locate the appropriate AAs and CSs. They interact with the CS to obtain, manage, and validate their credentials. The CS interacts directly with the AA in order to pass the End Users identity information, so the AA knows who they are dealing with. Once the identity information is known to the AA the user interacts directly with the AA for business transactions. Authorization is handled completely by the AA.

Governance is accomplished by managing the interaction between the AAs and CSs. The government will issue credentials to approved CSs and AAs, which will be validated before the End Users identity information is handed off.

[1] CSPs are sometimes referred to as Electronic Credential Providers (ECPs) in other documents.

There are three types of sessions discussed in the framework:

1. **Browser Session:** The period of time the End Users browser is open. The browser session begins when the user opens their browser and ends when it is closed. All session cookies are terminated when the Browser Session ends.
2. **Authentication Session:** The period of time that a user remains trusted after the user authenticates. A CS typically does not require a user to re-authenticate for every page they request; they continue to be trusted for some period of time after each authentication. The allowed period between re-authentication is referred to as the Authentication Session.
3. **Agency Session:** The period of time the AA will trust a user before they are handed off to the CS for re-authentication. AAs do not have access to Authentication Session information; they must maintain their own session with a user and decide how long a user remains authenticated once they have started their transaction.

The eAuthentication initiative has defined four assurance levels[2] to accommodate varying levels of risk. For the near term levels 1 and 2 will utilize credentials based on Personal Identification Numbers (PINs) and passwords. Levels 3 and 4 will use PKI based credentials. The principle reason for this distinction is that the higher assurance levels require[3] a cryptographic binding between the authentication and transaction, which is currently only widely available using client certificates over SSL or TLS.

PIN/password based systems and PKI systems are inherently different, both in terms of capabilities and the maturity of standards. The technical approach for the lower assurance levels is therefore different from the approach for higher assurance levels. Section 3.1 describes the approach for lower assurance levels, section 3.2 describes the approach for higher assurance levels.

3 Lower Assurance Levels

Currently there are a number of standards based schemes for federation at various stages of maturity, including Liberty, WS-Federation, SAML, and Shibboleth. Any of these schemes could be used to meet the lower assurance requirements of eAuthentication. It is unclear which of these schemes will become dominant in the market and it is quite possible more that one will be in common use. The following sections describe the framework for lower assurance authentication, show where the federation schemes fit in, and then describe how multiple schemes can be leveraged simultaneously.

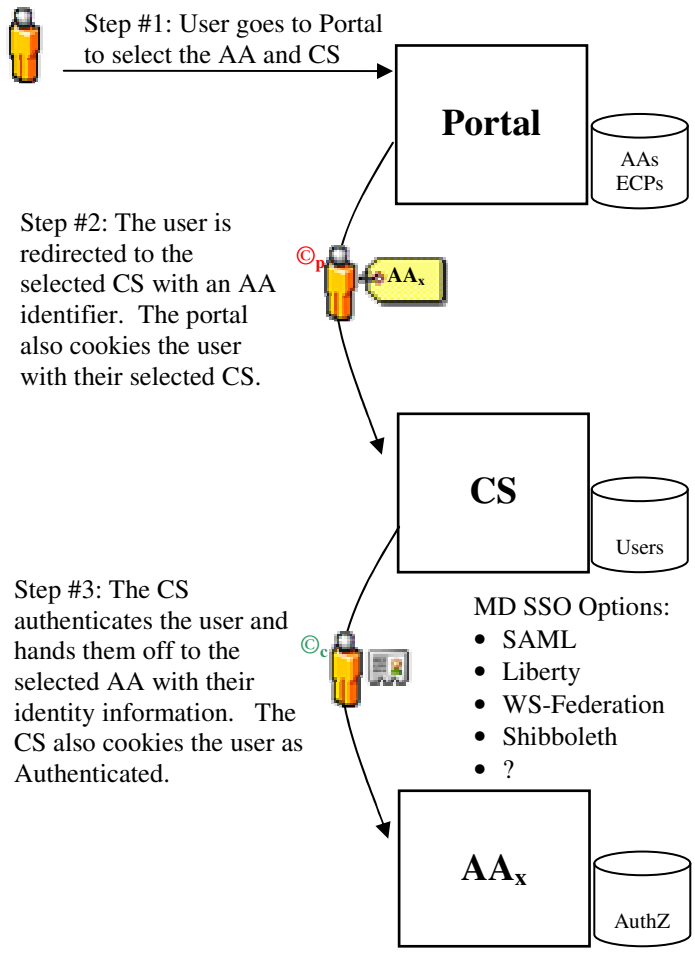
[2] Reference eAuthentication Guidance
[3] Reference NIST document

3.1 Base Case

The Base Case is the foundation of the framework, all other use cases build on and expand this case. Figure 2 depicts the sequence of events for the Base Case. In the first step the user goes to portal and selects the AA. The portal then presents the user with a list of CSs with appropriate assurance levels. Once the user selects their CS they are redirected to the CS with an identifier for the AA they have selected, shown in step 2. As part of this redirect the portal gives the user a session cookie indicating which CS the user has selected, which will remain for the duration of the Browser Session. This cookie is used to accommodate single sign-on in later transactions. The user then authenticates to the CS directly, and the CS assigns a session cookie to manage the Authentication Session. The final step is for the CS to pass the authenticated user on to the AA along with their identity, allowing the AA to manage transactions and authorization. Typically the AA will assign a cookie to manage the Agency Session.

Since the hand-off to the AA includes the identity of the user some Personally Identifiable Information (PII) is likely to be included. The CS may adjust the PII made available to a given AA based on the user's preferences, their privacy policies, or by prompting the user before the hand-off. eAuthentication interface specifications will specify the minimum set of identity attributes required for all hand-offs.

Figure 2: Base Case



The hand-off from the ECP to the AA shown in step #3 is a classic case of Multi-Domain Single Sign-On (MD SSO); a user authenticated in one domain (the CS) needs to become known to another domain (the AA) without re-authenticating. This hand-off is where the various federation schemes can be used. Currently the SAML, Liberty, Shibboleth, and WS-Federation schemes all provide mechanisms for MD SSO. Other standards based mechanisms are likely to become available and existing schemes are likely evolve.

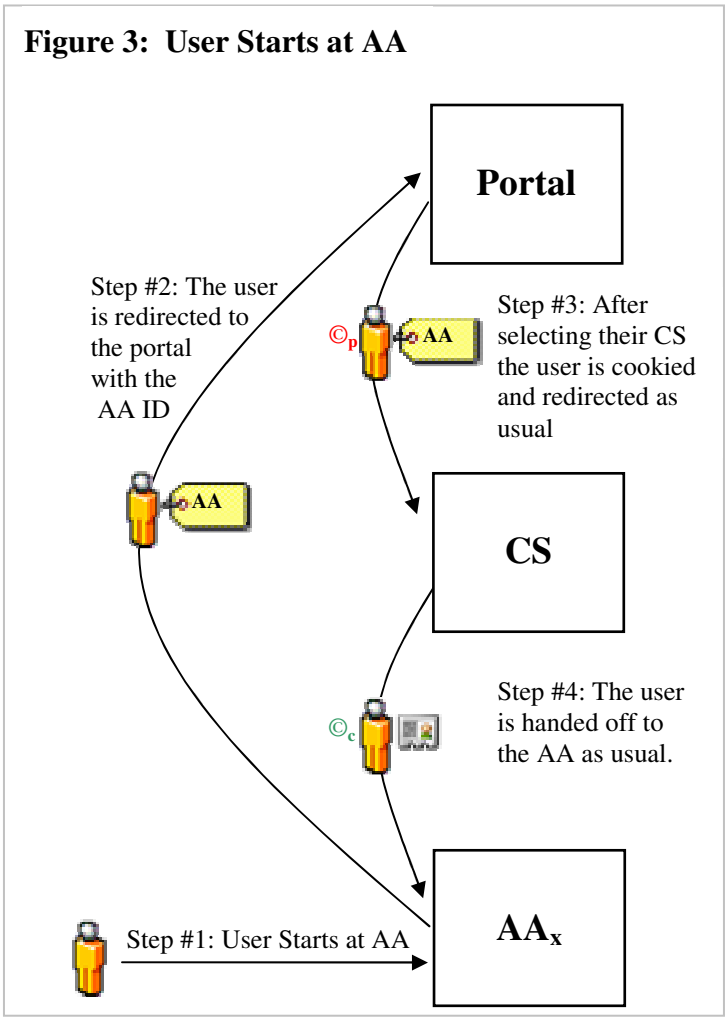
The uncertain future of industry standards is isolated to this final step of the Base Case. Section 3.2 discusses how various schemes can co-exist within the framework, and how graceful migration away from dying schemes is accomplished. The following sections describe additional use cases within the framework.

3.2 Starting at the Agency Application

It is unlikely that all users will start at the Portal in all cases. Figure 3 depicts the sequence of events for users that start at the AA. All applications in the architecture must be configured to redirect any unauthenticated user to the portal when they attempt to access a protected resource. When the user is redirected the identifier for the AA is included and passed to the portal, shown in step 2. The portal does not have to ask the user to select an application; it can simply display a list of appropriate credential services for selection by the user. If the user has previously authenticated during this session the portal cookie will provide the portal with the CS previously selected by the user, allowing the portal to immediately redirect the user without any user interaction. If the assurance level of the previously selected CS is insufficient for the AA requested the portal can notify the user and allow them to select an alternative CS.

Once the user has been redirected to the CS the sequence continues as described above in the base case. The user authenticates to the CS and is handed off to the originating AA using one of the MD SSO schemes. If the user has previously authenticated during this session then the CS cookie can be used to determine the users identity and the CS can initiate the hand-off without any user interaction, completing the single sign on sequence.

Figure 3: User Starts at AA



The combination of the Portal cookie and the CS cookie provide the mechanism for architecture-wide single sign-on, irregardless of the MD SSO scheme used. Once a user has authenticated the first time subsequent visits to other applications during the session will not require re-authentication. A user moving from one application to another will automatically become known to other applications once they have authenticated the first time. The portal cookie allows the user to be redirected to the CS without user interaction, and the CS cookie allows the user to be passed to the AA without interaction, providing seamless and invisible single sign-on.

Since the CS is directly involved in the single sign-on they have an opportunity to intervene if the user has opted out of single sign-on, or if their privacy policies prevent it. The management of these user preferences is encapsulated along with other identity management issues at the CSP; there is no need for a government-wide repository of these preferences.

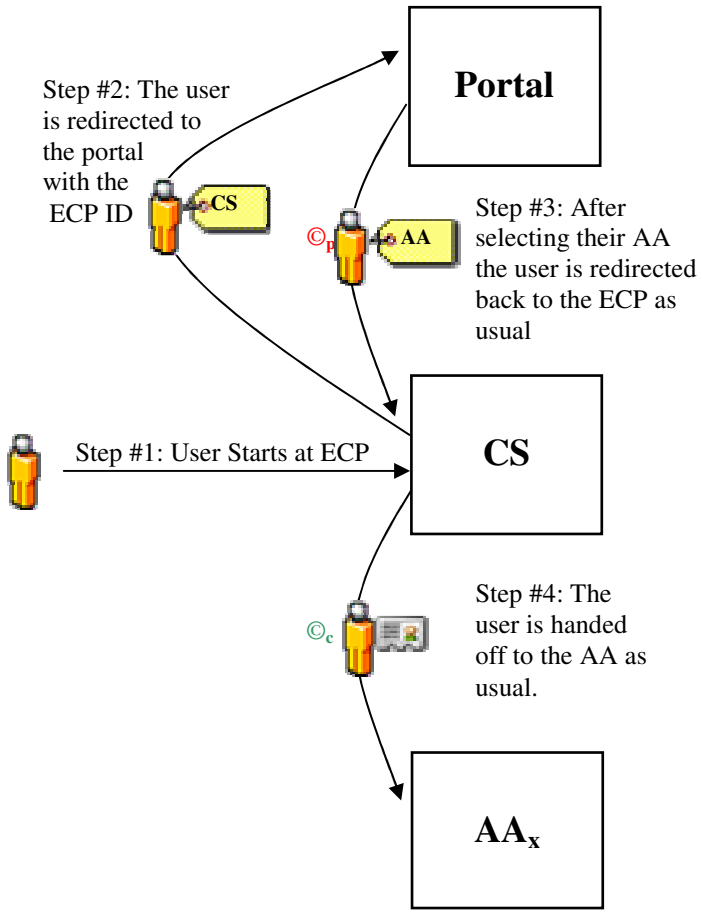
3.3 Starting at the Credentialing Service

In some cases the user may begin their session at the CS. For example, a bank that is integrated with eAuthentication may provide a link to the portal informing the user that their credential can be used to conduct government business. A user that is already authenticated to the bank conducting business may select the link to see what applications are available.

Figure 4 depicts the sequence of events for this case. The user starts at the CS and selects a link to the portal which includes a CS identifier, shown in step 1. The portal then presents a list of applications that can be accessed using the CS, as well as some indication that other applications may be available for higher assurance credentials. When the user selects an application they are redirected back to the originating CS with the AA identifier as shown in step 3. The sequence continues as described in the base case; after authenticating to the CS the user is handed off to the AA as shown in step 4.

This functionality allows CSs to advertise the utility of their credential, increasing the value proposition for CSPs. It also opens up every CS as a channel to advertise the availability of various agency applications.

Figure 4: User Starts at ECP



This use case illustrates the flexibility of the portal. Aside from supporting single sign-on the principle function of the portal is to help the user select their CS and AA. If the user implicitly makes one of those selections the framework allows the portal to avoid redundant user interaction. This capability reduces the required click count and generally simplifies the user experience.

See Appendix A for more use cases leveraging the flexibility of the portal.

3.4 Multiple Protocol Support

The description of the base case specified a role for schemes like Liberty, SAML, Shibboleth, and WS-Federation. These schemes specify protocols and standards for federated identity, mechanisms for different entities to share identities without requiring the user to manage multiple accounts. In the framework presented above those schemes are used in the final step of the base case, the hand-off of the user from the credential service to the application. This is also referred to as Multi-Domain Single Sign-On (MD SSO), where a user who has authenticated to one domain (the CS) becomes known to another domain (the AA) without re-authenticating.

It is currently unclear which of these schemes will become dominant in the market. They all represent different philosophies and approaches, and are at different levels of maturity. The only thing we know for certain is that the future is unpredictable. The architectural framework be designed to be durable and flexible, avoiding too much reliance on a single standard that may or may not survive. Even if a single standard had clear advantages today the technical approach should allow for graceful migration as new standards emerge or existing standards evolve. These different schemes also provide additional functionality beyond simple authentication, some of which may be necessary for some communities and useless for others. A successful government-wide authentication scheme should allow for the possibility of multiple schemes interoperating in a single architecture.

The encapsulation of decisions at the portal provides and opportunity for multiple schemes to coexist gracefully. The first step for the user at the portal is to select the application they wish to use, allowing the portal to present a list of appropriate credential services. The list of credential services should be based on compatible assurance levels, but could also be based on compatible MD SSO schemes. If the user selects an agency application that supports SAML and Liberty, then only credential services supporting one of those schemes would be provided to the user.

This approach does introduce a problem, it may be impossible for an agency application to leverage all the credential services if there are too many protocol disparities. This is a problem that will have to be monitored closely by the initiative or a principle vision of the initiative may be lost. There are several solutions available to mitigate this problem:

1. multiple protocol support by CSPs
2. multiple protocol support by AAs
3. eAuthentication sponsored Protocol Translators.

Encouraging CSs to support multiple protocols may be a viable solution if there are not many CSs or if there is a sufficient business case to warrant the investment. A credential issued by a service that supports more schemes is certainly more valuable because it would be usable by more sites, but depending on the business model the value may not warrant the investment. It is also possible that COTS employed by the CS would natively support multiple protocols, which would not be surprising for closely related schemes such as SAML and Liberty.

Encouraging AAs to support multiple protocols may also be viable for similar reasons. If their COTS natively support multiple schemes it would make sense, or if the perceived value of being able to rely on more credential services warranted the additional investment. In general this is probably an unrealistic option given the typical agility of agency applications.

eAuthentication sponsored Protocol Translators (PTs) are the most viable solution to protocol disparities. At a high level the role of the translator is simple; it acts as an intermediary for incompatible CSs and AAs by supporting multiple MD SSO schemes. For example, a PT that

supported WS-Federation and Liberty would allow the two communities to have interoperable authentication while enjoying whatever other benefits each scheme provides to each community.

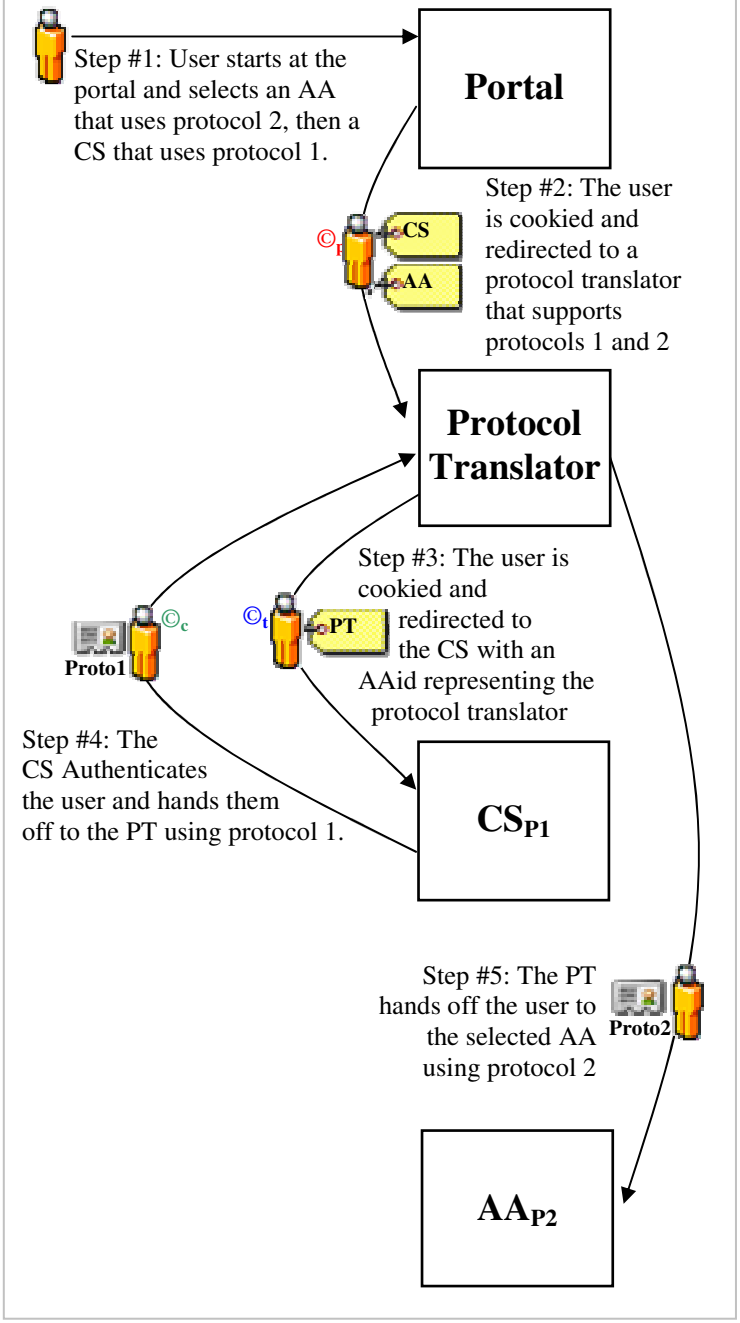
The protocol translators simply pass through identity information based on standards already adopted in the architecture. Multiple translators could be deployed to increase availability and end user privacy. There is also no need for AAs or CSs to engage in any special integration for translators. The translators appear to be any other CS from the AA perspective and any other AA from the CS perspective. Organizations that have invested in one of the supported architectures will be able to use their existing systems so long as the translators are available.

The following section depicts the sequence of events for an authentication that involves a protocol translator.

3.4.1 Protocol Translators

Figure 5 shows the how a translator fits into the framework. The user starts at the portal as usual to select their CS and AA as described in the use cases above. When the user selects their AA the portal provides a list of CSs that have an appropriate assurance level, have MD SSO schemes compatible with the AA, or have MD SSO schemes compatible with an appropriate protocol translator. If the CS and AA are directly compatible the session continues as described in the base case. If the translator is required the user is redirected to the translator with the AA and ECP identifiers as shown in step #2. The translator then cookies the user with both identifiers and redirects the user to the CS with an AA identifier that represents the translator. The CS performs the same functions as any other use case, authenticating and handing off the user, shown in step 4. The translator now has the identity information for the user and initiates a hand-off to the AA using the second protocol.

Figure 5: Protocol Translator



The translator cookie records the destination AA identifier, so the translator knows where to hand-off the user once they are returned from the CS.

Since the translator does not interact with the user it's role is completely transparent. The CS interacts with the translator as if it were any other AA, so no additional functionality is required by AAs to interface with translators. The AA interacts with the translator as if it were any other CS, so no additional functionality is required by the CS to interface with translators. Only the portal configuration and the translator itself are required to bridge the gap between multiple schemes.

3.5 Management over Time

The protocol translators allow for multiple schemes to co-exist within the framework, but the eAuthentication initiative must carefully govern the introduction of new schemes. The translators do add complexity to the architecture and establish an additional point of failure in transactions, so their use should be minimized. Ideally only a small number of schemes would exist in the architecture at any given time and protocol translators would be phased out over time as various components adopt dominant schemes.

Figure 6 depicts the lifecycle for adoption of new schemes. As new schemes emerge that meet eAuthentication requirements they are assessed for the availability of interoperable COTS, then piloted on a small scale. If the pilots are successful then existing components can either migrate to support of the new scheme or the initiative can deploy translators. The translators eliminate the need for every component to migrate at the same pace, slower components can rely on translators until they are ready. Components which have adopted the new schemes can begin to use them immediately, enjoying whatever other features they may offer without losing authentication interoperability with the rest of the eAuthentication components.

Figure 6: Scheme Adoption

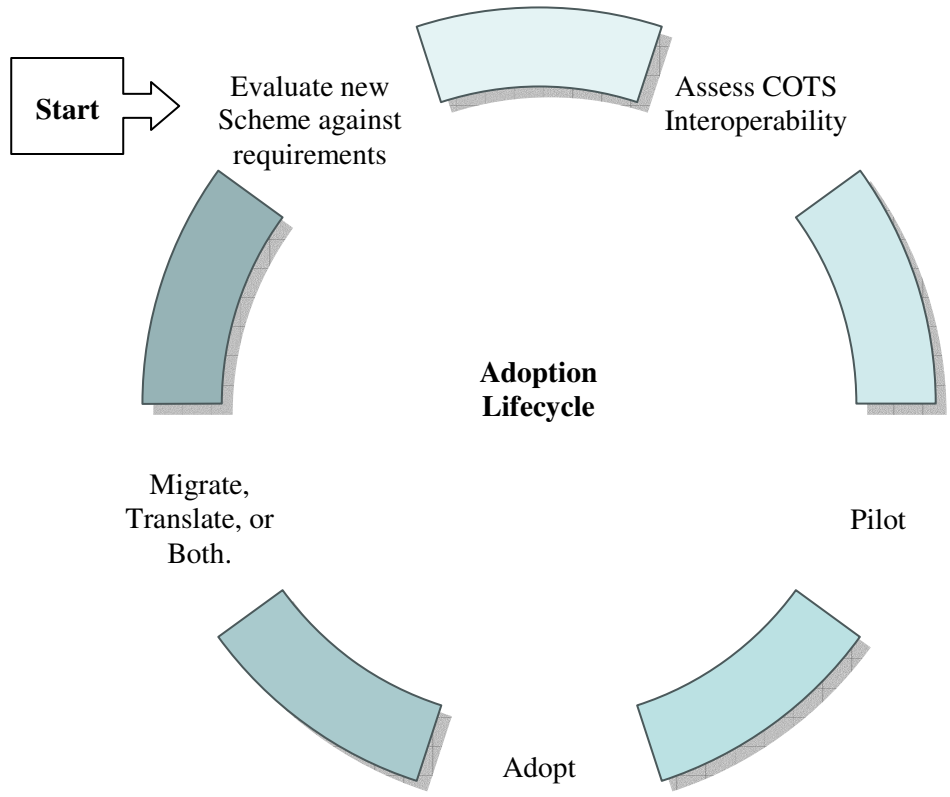


Figure 7: Scheme Adoption Process

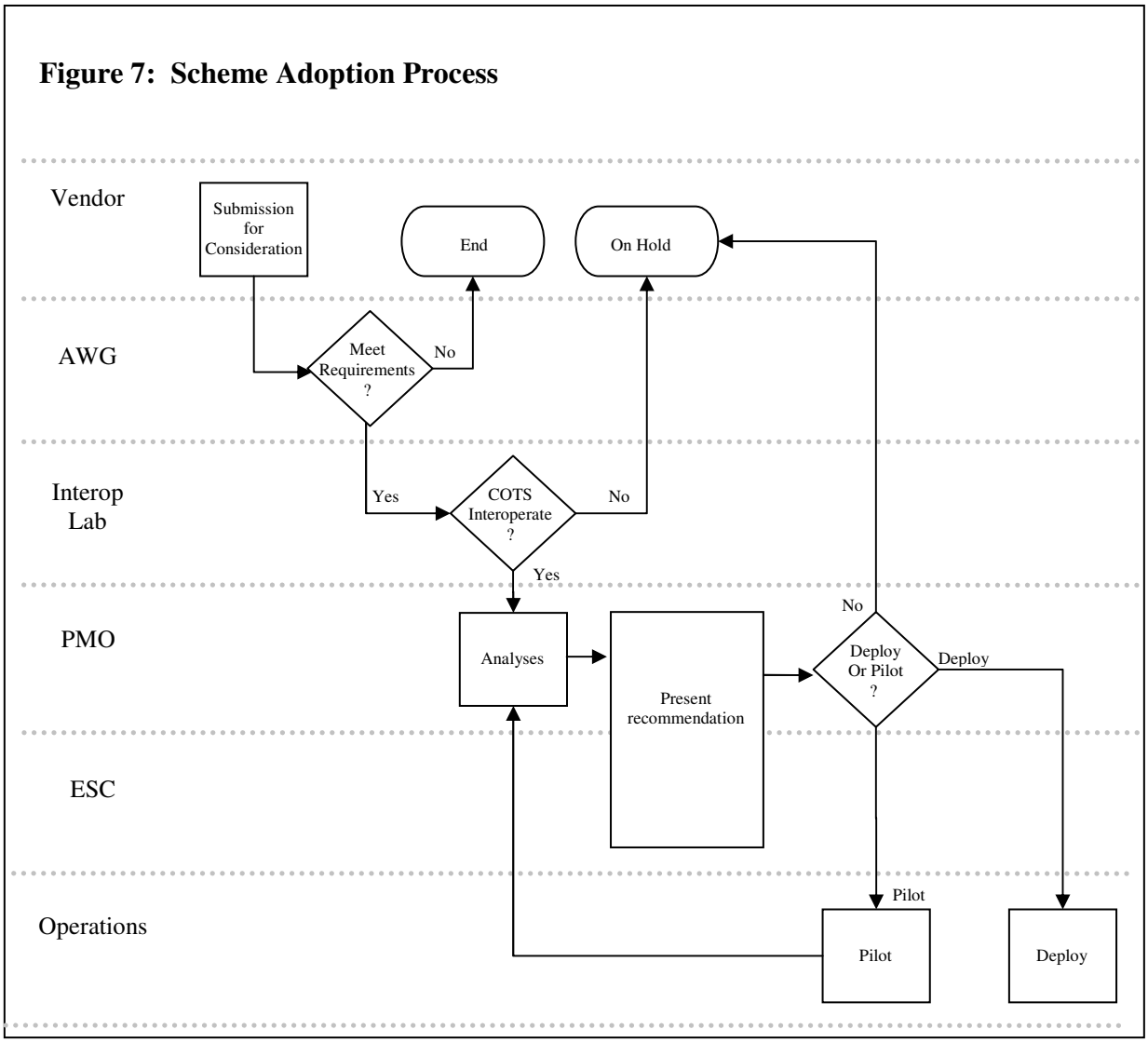


Figure 7 shows a process for adopting new schemes. Once new schemes are submitted they are evaluated against eAuthentication requirements by the Architecture Working Group (AWG). The AWG would also determine how the scheme needed to be constrained to meet federal needs, providing specifications for a federal profile of the scheme. Next the interoperability lab will assess the state of COTS interoperability and provide analyses to the PMO. If sufficient interoperable COTS exist and the scheme offers sufficient benefit to the government the Executive Steering Committee (ESC) will decide whether to deploy or pilot the scheme within the architecture.

4 Higher Assurance Levels

PKI based credentials offer considerable advantages for authentication. They are capable of higher assurance transactions and can be validated using only public information. The standards for PKI are also more mature and more widely used than the emerging standards for federated PIN/Password based identity management.

The Federal PKI (FPKI) employs a Bridge Certificate Authority (BCA) to harmonize policies and procedures for Certificate Authorities (CAs). The eAuthentication initiative has deferred assessment and governance of PKI based credential services to the FPKI PA, the governing body for the BCA. Additional information on the FPKI is available at http://www.cio.gov/fpkipa/.

The eAuthentication technical approach for accepting PKI based credentials is based on providing mechanisms for AAs to validate certificates. The following sections describe the various use cases for certificate validation services.

4.1 Certificate Validation Service

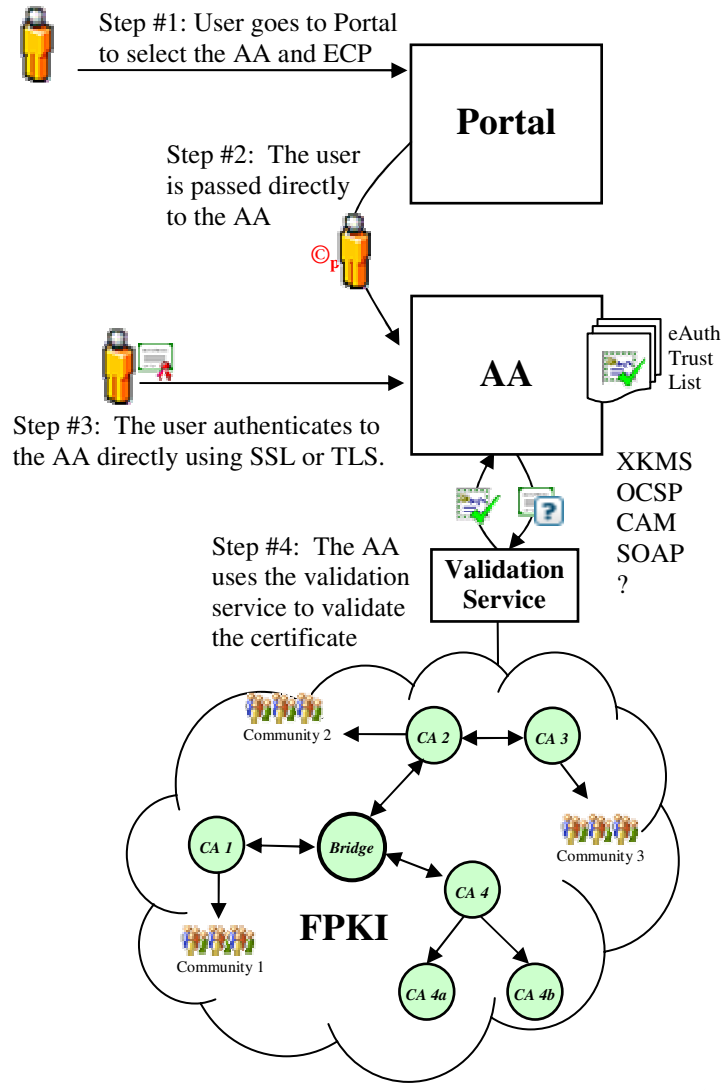
The eAuthentication will offer a certificate validation service to agency applications. Figure 8 depicts the use of the service for authentication. The user starts at the portal as usual, but is passed directly to the AA for authentication. There is no need for the user to be sent to the CS because TLS and SSL allow the user to authenticate using their certificate without revealing any secret information. The AA authenticates the user in step 3, then delegate's validation of the certificate to the validation service in step 4.

To the greatest extent possible the validation service will be comprised of COTS products using standard protocols. NIST has established requirements for certificate path validation that can be used by the eAuthentication interoperability lab to determine appropriate products and interface specifications.

Over time the validation service may support multiple products and standards, but the functionality will remain the same. Again, the approach is a framework showing where appropriate standards can be adopted as they mature.

The eAuth trust list must be used by the AA in the TLS/SSL connection. The protocol requires the web server to present a list of acceptable certificate authorities to the browser during the TLS/SSL handshake.

Figure 8: FPKI



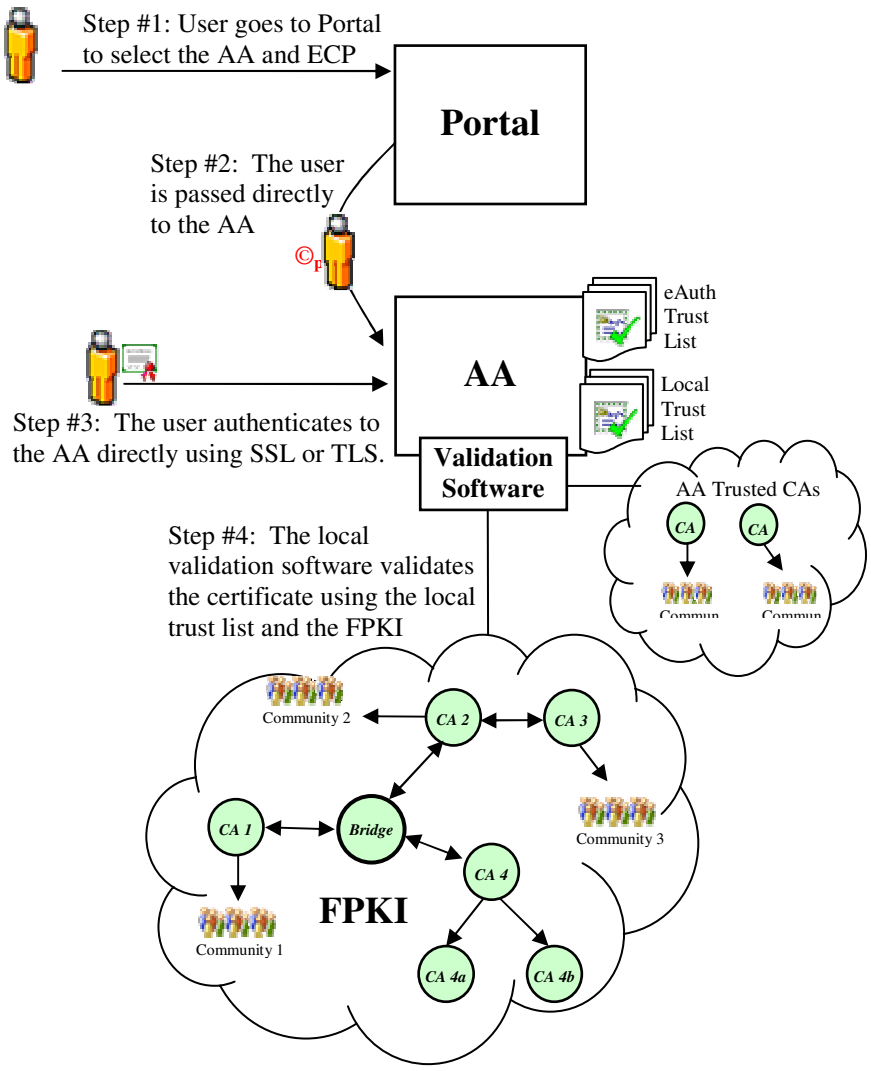
4.2 Local Validation

In some cases agencies may wish to perform certification validation locally. For example, if an agency has elected to trust certificate authorities that are not part of the eAuthentication trust list they would have to maintain their own local trust list.

The eAuthentication initiative will support these agencies by providing validation software that can be run locally and integrated with custom trust lists. The initiative will perform software evaluation based on the FPKI requirements established by NIST and make applicable software available to agencies.

Figure 9 depicts this use case. The user starts at the portal and validates to the AA as described above. The AA then uses locally installed validation software that has been integrated with their custom trust list to validate credentials directly. Communication with the validation service is not required.

Figure 9: FPKI



The Validation Software must still be capable of operating within the FPKI to ensure other eAuthentication credentials can be accepted. It will be configured locally to extend that trust to locally configured credentials to meet the special needs of some AAs.

4.3 High Assurance Credentials at Lower Assurance Applications

One of the requirements for eAuthentication is that credentials should be usable any agency application with an equal or lower assurance level. That implies that PKI credentials should be usable at lower assurance applications. In order to avoid the need for lower assurance applications to validate certificates the initiative will deploy a protocol translator that supports certificate validation and the dominant MD-SSO schemes in use by lower assurance applications.

Figure 10: PKI credentials for low assurance AAs

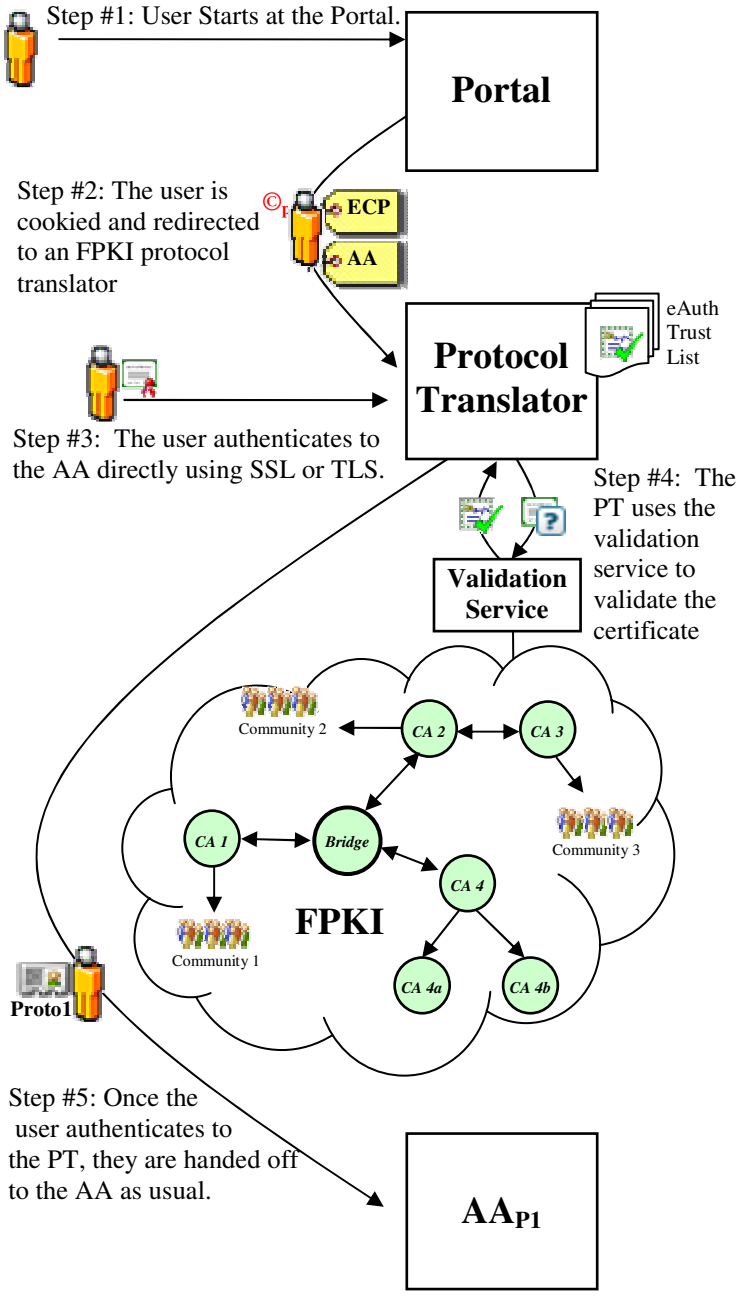


Figure 10 shows the sequence of events for a user with a PKI credential accessing a lower assurance application. In step 1 the user begins at the portal and selects their CS and AA as usual. The portal then hands off the user to the protocol translator with the CS and AA identifiers as shown in step 2. In step 3 the user authenticates to the translator using their certificate. Next the translator uses the validation service to validate the certificate before handing off the user to the AA in step 5.

The AA does not have to deploy and special capabilities to leverage the protocol translator. The translator interacts with the AA like any other CS in the Base Case. Decisions on whether a translator is required are also encapsulated at the portal, further insulating the AA from any special requirements.

5 Implementation

The framework presented in this paper does not prescribe the specific standards currently employed. This document must be accompanied by further specification of adopted schemes that depict the architecture at a point in time. Each of the adopted schemes must be further accompanied by interface specifications that provide detailed technical specifications for how to use a given scheme within the framework. Current information will be maintained by the eAuthentication initiative and made available at their website, http://www.cio.gov/eAuthentication/.

AAs and CSs joining the eAuthentication community must select one of the adopted schemes to interoperate with other components in the architecture. The initiative will also provide a list of tested COTS products within each scheme that have proven interoperability according to federal standards and specifications. Additional agreements beyond the scope of this document are also required, interested parties should contact the eAuthentication PMO for more information.

Appendix A: Distributed Portal Functionality

1 Introduction

The AA and CS metadata stored at the portal could be shared with other entities. There is nothing sensitive about the information and no reason to keep it isolated at the portal. Other sites equipped with the information could assist users in the select of an AA or ECP. The portal's ability to process passed AA and CS ids enables other sites to add value without requiring redundant interaction with end users.

One example is for a credential service to present the end user with agency applications that will accept their credentials. CSs such as banks may be able to add value by suggesting agency applications that are relevant to a particular user or related to the business they are engaged in during a particular session. A CS that has downloaded the metadata about agency applications is considered portal enabled, ie it has the ability to present the user with applications that are accessible with their credential. The following sections describe use cases where other sits have been portal enabled.

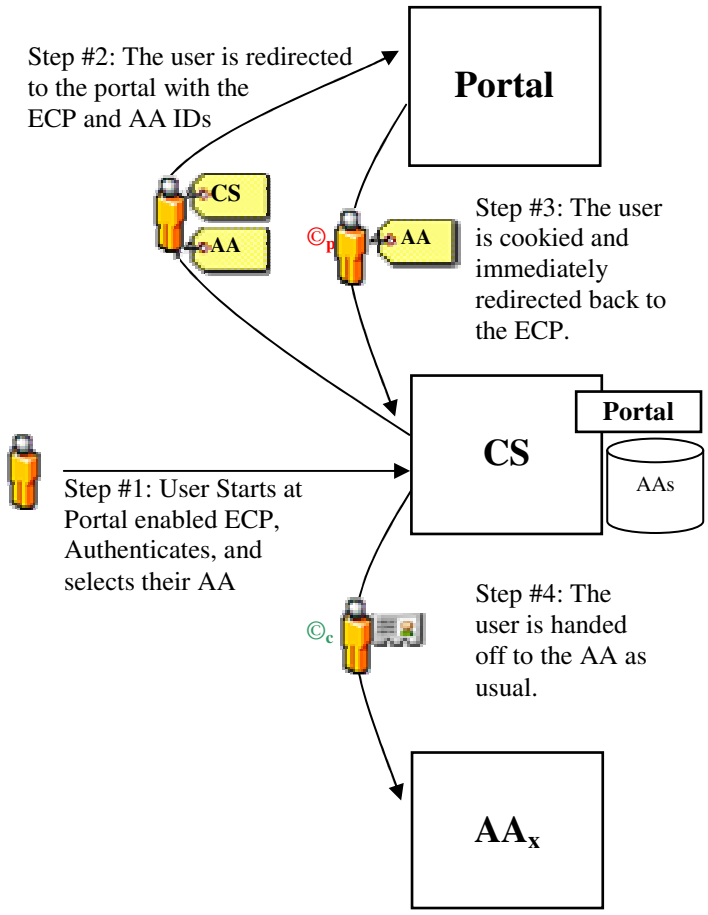
2 Portal Functions at the Credential Service

It is possible for a CSP to provide some portal functions in this framework. Figure 11 shows the sequence of events for this case. The user starts at their credential service, perhaps conducting routine business. The CS has integrated metadata on agency applications into their site and presents the user with a list of applications that can be accessed with their credential. When the user selects one of the applications the CS redirects the user to the portal with the application identifier and the CS identifier, shown in step 2. Since the user arrives at the portal with both identifiers there is no need for the portal to interact with the user, they are simply cookied and passed to the CS as shown in step 3. In step 4 the CS passes the user to the AA as described in the base case.

While it would be possible for the CS to initiate the hand-off to the AA directly, the user must be sent to the portal in order for single sign-on to work properly. If the CS passed the user directly to the AA then subsequent visits to other applications would not be automatically authenticated. Single sign-on requires the portal cookie as well as the CS cookie, so even when the user is not interacting with the portal they must be passed through it.

Explicit support for this scenario in the architecture encourages credential providers to advertise the availability of government applications. It also provides an easy mechanism for credential services to show off the value of their credential to their user base. The end user benefits from easier availability and access to government applications.

Figure 11: User Starts at portal capable ECP



3 Portal Functions at the Agency Application

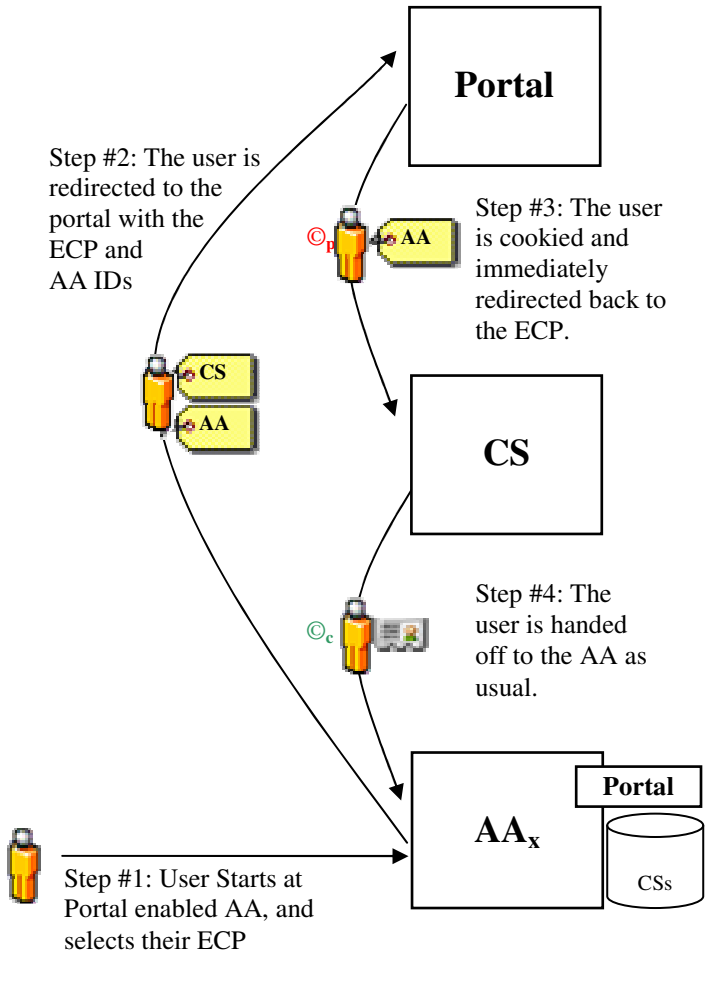
It is also possible for an agency application to provide some portal functionality. If an AA downloads the metadata for ECPs they could directly provide end users with a list of potential credential providers.

Figure 12 shows the sequence of events for the portal enabled AA case. The user starts at the agency application which has integrated the metadata for credential services. The application can then present the user with a list of appropriate credentials services directly from the AA site. Once the user selects their CS they are redirected to the portal with the AA and CS identifiers, shown in step 2. Once again the portal does not need to interact with the user, so they are simply cookied and passed along to the CS as described in the base case, shown in step 3. Finally, in step 4, the user is then authenticated and passed back to the AA as described in the base case.

This scenario is not recommended because it can interfere with single sign-on. If the user had already authenticated to a different AA earlier in their session, then accessed the portal enabled application, they would have to select their CS a second time at the portal enabled AA. If the AA simply redirected the user to the portal as described in figure 2 they would not be required to make the selection a second time. This scenario is presented because it may provide utility to some agencies in certain circumstances and requires no additional functionality in other architectural components.

If this was the users first authentication then subsequent access to other agency applications would provide single sign on.

Figure 12: User Starts at portal capable AA



4 Other Possibilities

The ability of the portal to accept incoming AA and CS identifiers supports a variety of scenarios that allow for flexibility in the user experience. For example, it would also be possible for a commercial portal to download all of the metadata and provide portal functionality, simply redirecting the user to the eAuthentication portal once the user made their selections. An industry association website could also integrate the metadata into their site providing members with easy access to industry related applications, again redirecting the user to the eAuthentication portal once the user made their selections. Agency websites could provide similar functionality, highlighting the applications provided by the agency. These scenarios and other are all possible and ultimately benefit the user and the government by increasing the exposure of eGovernment applications.

Figure 13: User Starts at another site

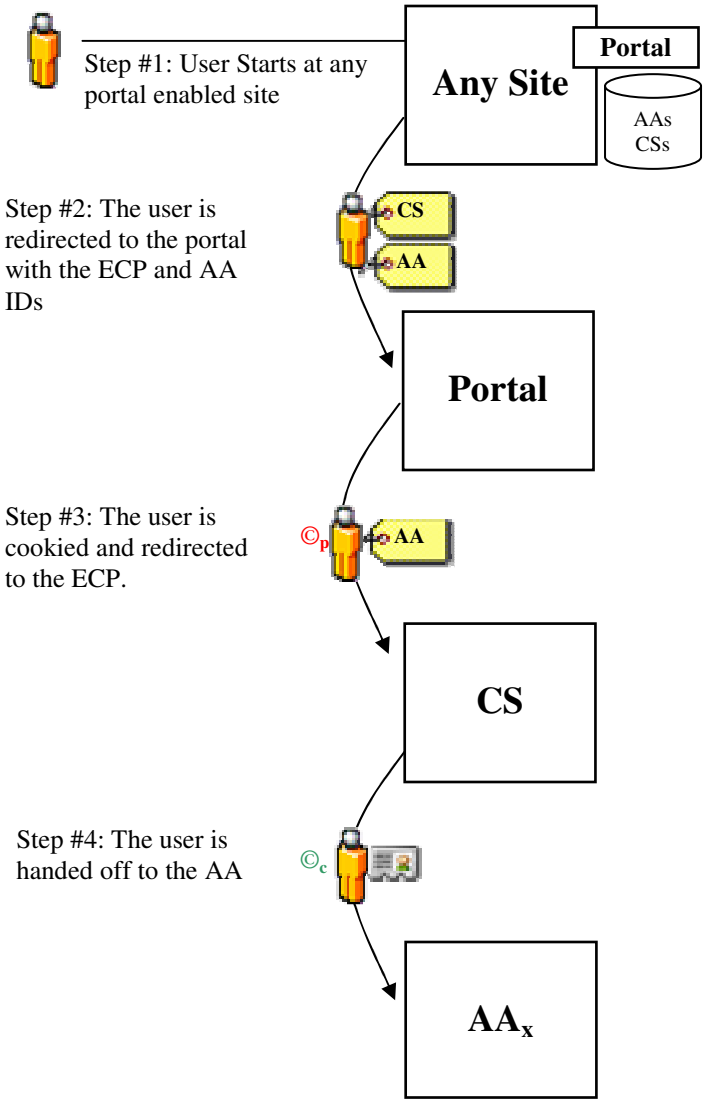


Figure 13 depicts the sequence of events for these scenarios. The user starts at any site which has integrated the portal metadata and makes their decisions. In step 2 they are redirected to the portal with the CS and AA identifiers as described in the previous cases. The portal can then immediately redirect the user the CS without any interaction as shown in step 3. The sequence then continues as described in the base case; where the user authenticates to the CS and then is passed to the AA in step 4.